

AMENDMENTS TO THE CLAIMS

Claims 1, 4, 13, 19-20, 22-26, 28-36, 42-43, 45-46, 49, 51-55, 59-61, 63-64, 67-72, and 74 are amended.

1. (Currently amended) A computer-implemented user interface configuration method, for configuring the user interface of a software application and user interface of an operating system of a computer system, comprising:

storing a plurality of application markers, each application marker indicat-

ing a user interaction with one of the application programs,

wherein the application markers include markers for the plurality

of different applications;

storing a plurality of operating system markers, each operating system

marker indicating a user interaction with the operating system;

determining a score as a function of the operating system markers and the

application markers;

~~detecting~~ determining a user proficiency level with respect to ~~[[a]]~~ the user

interface ~~of the software application and user interface of the oper-~~

~~ating system, based on user behavior with respect to the user inter-~~

~~face~~ based upon the score; and

automatically configuring at least one functional component of the user in-

terface of the software application and at least one functional com-

ponent of the user interface of the operating system responsive to

the ~~detected~~ user proficiency level.

2. (Previously Presented) The method of claim 1, wherein automatically configuring the at least one functional component of the user interface comprises:

selecting at least one configuration option from a plurality of configuration options.

3. (Previously Presented) The method of claim 1, wherein automatically configuring the at least one functional component of the user interface comprises at least one selected from the group consisting of:

enabling access to a functional user interface element;
disabling access to a functional user interface element; and
changing an appearance of a functional user interface element.

4. (Currently Amended) The method of claim 1, wherein automatically configuring the at least one functional component of the user interface comprises ~~at least one selected from the group consisting of:~~

providing a set of functions including:

enabling access to a command;
disabling access to a command;
changing an appearance of a command;
enabling access to a menu;
disabling access to a menu;
changing an appearance of a menu;
enabling access to a button;
disabling access to a button;
changing an appearance of a button;
enabling access to a shortcut; and

disabling access to a shortcut; and
selecting at least one of the provided functions to configure the functional
component.

5. (Cancelled).
6. (Cancelled).
7. (Cancelled).
8. (Cancelled).
9. (Cancelled).
10. (Cancelled).
11. (Original) The method of claim 1, further comprising:
outputting a notification of a change to user interface configuration.
12. (Original) The method of claim 1, further comprising:
outputting a notification of at least one newly enabled user interface feature.
13. (Currently amended) The method of claim 1, wherein ~~detecting~~ determining
the user proficiency level and automatically configuring the user interface are performed
responsive to a trigger event.

14. (Original) The method of claim 13, wherein the trigger event comprises user input requesting user interface configuration.

15. (Original) The method of claim 13, wherein the trigger event comprises application startup.

16. (Original) The method of claim 13, wherein the trigger event comprises system startup.

17. (Original) The method of claim 13, wherein the trigger event comprises a change in user behavior with respect to the user interface.

18. (Original) The method of claim 13, wherein the trigger event comprises user logon.

19. (Currently amended) The method of claim 1, wherein ~~detecting~~ determining the user proficiency level and automatically configuring the at least one functional component of the user interface are performed periodically.

20. (Currently amended) The method of claim 1, wherein ~~detecting~~ determining the user proficiency level comprises reading a stored user proficiency level derived from at least one marker.

21. (Original) The method of claim 20, wherein the marker indicates historical usage of the user interface.

22. (Currently amended) The method of claim 1, wherein ~~detecting~~ determining the user proficiency level comprises ~~detecting~~ determining whether a user interface element has been used.

23. (Currently amended) The method of claim 1, wherein ~~detecting~~ determining the user proficiency level comprises ~~detecting~~ determining whether a user interface element has been used a number of times exceeding a predetermined threshold.

24. (Currently amended) The method of claim 1, wherein ~~detecting~~ determining the user proficiency level comprises ~~detecting~~ determining a total amount of time spent by a user using an application.

25. (Currently amended) The method of claim 1, wherein ~~detecting~~ determining the user proficiency level comprises ~~detecting~~ determining how many applications are open concurrently.

26. (Currently amended) The method of claim 1, wherein ~~detecting~~ determining the user proficiency level comprises ~~detecting~~ determining a historical average number of concurrently open applications.

27. (Cancelled).

28. (Currently amended) The method of claim 1, wherein ~~detecting~~ determining the user proficiency level comprises ~~detecting~~ determining how many windows are open concurrently.

29. (Currently amended) The method of claim 1, wherein ~~detecting~~ determining the user proficiency level comprises ~~detecting~~ determining a historical average number of concurrently open windows.

30. (Currently amended) The method of claim 1, wherein ~~detecting~~ determining the user proficiency level comprises ~~detecting~~ determining a user-specified preference indicating a proficiency level.

31. (Currently amended) The method of claim 1, wherein ~~detecting~~ determining the user proficiency level comprises ~~detecting~~ determining web page visitation patterns.

32. (Currently amended) The method of claim 1, wherein ~~detecting~~ determining the user proficiency level comprises ~~detecting~~ determining historical usage of secure web pages.

33. (Currently amended) The method of claim 1, wherein ~~detecting~~ determining the user proficiency level comprises ~~detecting~~ determining historical usage of web pages having active content.

34. (Currently amended) The method of claim 1, wherein:

~~detecting~~ determining the user proficiency level comprises ~~detecting~~ determining the user proficiency level with respect to a user interface component less than the entire user interface; and
automatically configuring the at least one functional component of the user interface comprises automatically configuring the user interface component without altering the configuration of the remainder of the user interface.

35. (Currently amended) The method of claim 1, wherein:
~~detecting~~ determining the user proficiency level comprises ~~detecting de-~~
termining the user proficiency level with respect to an application;
and
automatically configuring ~~the~~ at least one functional component of the
user interface comprises automatically configuring the user inter-
face for the application.
36. (Currently amended) The method of claim 1, further comprising:
responsive to user behavior with respect to the user interface, storing a
marker indicating a user proficiency level;
and wherein ~~detecting~~ determining the user proficiency level comprises
reading the stored marker.
37. (Original) The method of claim 36, wherein:
storing the marker is performed by a first application; and
reading the stored marker is performed by a background process.
38. (Original) The method of claim 36, wherein:
storing the marker is performed by a first application; and
reading the stored marker is performed by a second application different
from the first application.
39. (Original) The method of claim 36, wherein:
storing the marker is performed by an operating system; and
reading the stored marker is performed by the operating system.

40. (Previously Presented) The method of claim 39, wherein:
automatically configuring the at least one functional component of the
user interface comprises modifying functional user interface elements that are supplied to a plurality of applications.
41. (Original) The method of claim 36, wherein:
storing the marker is performed by an operating system; and
reading the stored marker is performed by an application.
42. (Currently amended) The method of claim 1, wherein ~~detecting~~ determining
the user proficiency level comprises retrieving a plurality of stored markers and aggregating the retrieved markers to derive a proficiency level.
43. (Currently amended) The method of claim 1, further comprising:
responsive to user behavior with respect to the user interface, storing a
plurality of markers;
and wherein ~~detecting~~ determining the user proficiency level comprises retrieving at least a subset of the stored markers and aggregating the retrieved markers to derive a proficiency level.
44. (Original) The method of claim 1, further comprising:
accepting user input overriding the user interface configuration and specifying a desired configuration; and
responsive to the user input, configuring the user interface according to the desired configuration.

45. (Currently amended) The method of claim 1, wherein:

~~detecting~~ determining a user proficiency level with respect to a user interface comprises ~~detecting~~ determining a user proficiency level with respect to a user interface of a web-resident application being run from a client machine; and

automatically configuring the at least one functional component of the user interface comprises automatically configuring at least one functional user interface element for the web-resident application.

46. (Currently amended) A computer program product for configuring a user interface of a software application and user interface of an operating system of a computer system, comprising:

a computer-readable medium; and

computer program code, encoded on the medium, which the code is executed by the computer system, for:

storing a plurality of application markers, each application marker indicating a user interaction with one of the application programs, wherein the application markers include markers for the plurality of different applications;

storing a plurality of operating system markers, each operating system marker indicating a user interaction with the operating system;

determining a score as a function of the operating system markers and the application markers;

~~detecting~~ determining a user proficiency level with respect to [[a]]
the user interface of the software application and user inter-
face of the operating system, based on user behavior with-
respect to the user interface based upon the score; and
automatically configuring at least one functional component of the
user interface of the software application and at least one
functional component of the user interface of the operating
system responsive to the ~~detected~~ user proficiency level.

47. (Previously Presented) The computer program product of claim 46, wherein the computer program code for automatically configuring the at least one functional component of the user interface comprises computer program code for:

selecting at least one configuration option from a plurality of configuration options.

48. (Previously Presented) The computer program product of claim 46, wherein the computer program code for automatically configuring the at least one functional component of the user interface comprises at least one selected from the group consisting of:

computer program code for enabling access to a functional user interface element;

computer program code for disabling access to a functional user interface element; and

computer program code for changing an appearance of a functional user interface element.

49. (Currently Amended) The computer program product of claim 46, wherein the computer program code for automatically configuring the at least one functional component of the user interface comprises ~~at least one selected from the group consisting of:~~

- computer program code for enabling access to a command;
- computer program code for disabling access to a command;
- computer program code for changing an appearance of a command;
- computer program code for enabling access to a menu;
- computer program code for disabling access to a menu;
- computer program code for changing an appearance of a menu;
- computer program code for enabling access to a button;
- computer program code for disabling access to a button;
- computer program code for changing an appearance of a button;
- computer program code for enabling access to a shortcut; and
- computer program code for disabling access to a shortcut.

50. (Cancelled).

51. (Currently amended) The computer program product of claim 46, wherein the computer program code for ~~detecting~~ determining the user proficiency level and automatically configuring the at least one functional component of the user interface comprises computer program code for performing the ~~detecting~~ determining and configuring steps responsive to a trigger event.

52. (Currently amended) The computer program product of claim 46, wherein the computer program code for ~~detecting~~ determining the user proficiency level and automatically configuring the at least one functional component of the user interface comprises computer program code for performing the ~~detecting~~ determining and configuring steps periodically.

53. (Currently amended) The computer program product of claim 46, wherein the computer program code for ~~detecting~~ determining the user proficiency level comprises computer program code for reading a stored user proficiency level derived from at least one marker.

54. (Currently amended) The computer program product of claim 46, wherein:
the computer program code for ~~detecting~~ determining the user proficiency level comprises computer program code for ~~detecting~~ determining the user proficiency level with respect to a user interface component less than the entire user interface; and
the computer program code for automatically configuring the at least one functional component of the user interface comprises computer program code for automatically configuring the functional user interface component without altering the configuration of the remainder of the user interface.

55. (Currently amended) The computer program product of claim 46, wherein:
the computer program code for ~~detecting~~ determining the user proficiency level comprises computer program code for ~~detecting~~ determining the user proficiency level with respect to an application; and

the computer program code for automatically configuring the at least one functional component of the user interface comprises computer program code for automatically configuring the user interface for the application.

56. (Original) The computer program product of claim 46, further comprising:
computer program code for, responsive to user behavior with respect to the user interface, storing a marker indicating a user proficiency level;
and wherein the computer program code for detecting the user proficiency level comprises computer program code for reading the stored marker.

57. (Original) The computer program product of claim 46, wherein the computer program code for detecting the user proficiency level comprises computer program code for retrieving a plurality of stored markers and aggregating the retrieved markers to derive a proficiency level.

58. (Original) The computer program product of claim 46, further comprising:
computer program code for, responsive to user behavior with respect to the user interface, storing a plurality of markers;
and wherein the computer program code for detecting the user proficiency level comprises computer program code for retrieving at least a subset of the stored markers and aggregating the retrieved markers to derive a proficiency level.

59. (Currently amended) The computer program product of claim 46, wherein:

the computer program code for ~~detecting~~ determining a user proficiency level with respect to a user interface comprises computer program code for ~~detecting~~ determining a user proficiency level with respect to a user interface of a web-resident application being run from a client machine; and

the computer program code for automatically configuring the at least one functional component of the user interface comprises computer program code for automatically configuring at least one functional user interface element for the web-resident application.

60. (Currently amended) A system for configuring a user interface of a software application and user interface of an operating system of a computer system, comprising:

means for storing a plurality of application markers, each application marker indicating a user interaction with one of the application programs, wherein the application markers include markers for the plurality of different applications;

means for storing a plurality of operating system markers, each operating system marker indicating a user interaction with the operating system;

means executed by the computer system for determining a score as a function of the operating system markers and the application markers;

means executed by the computer system, for ~~detecting~~ determining a user proficiency level with respect to [[a]] the user interface of the software application or user interface of the operating system.

~~based on user behavior with respect to the user interface~~ based upon the score; and
means executed by the computer system, for automatically configuring at least one functional component of the user interface of the software application and at least one functional component of the user interface of the operating system responsive to the ~~detected~~ user proficiency level.

61. (Currently amended) A system for configuring a user interface of a software application and user interface of an operating system of a computer system, comprising:
a marker storage device for,
storing a plurality of application markers, each application marker indicating a user interaction with one of the application programs, wherein the application markers include markers for the plurality of different applications; and
storing a plurality of operating system markers, each operating system marker indicating a user interaction with the operating system;
a user proficiency level determiner, executed by the computer system and coupled to the marker storage device, for
determining a score as a function of the operating system markers and the application markers; and
detecting determining a user proficiency level with respect to [[a]]
the user interface of the software application or user interface of the operating system, based on user behavior with respect to the user interface based upon the score; and

a user interface configuration module, executed by the computer system
and coupled to the user proficiency level ~~detector~~ determiner, for
automatically configuring at least one functional component of the
user interface of the software application and at least one func-
tional component of the user interface of the operating system re-
sponsive to the ~~detected~~ user proficiency level.

62. (Original) The system of claim 61, wherein the user interface configuration module selects at least one configuration option from a plurality of configuration options.

63. (Currently amended) The system of claim 61, wherein the user interface configuration module ~~performs at least one selected from the group consisting of~~ comprises
program code for performing the functions of:

enabling access to a functional user interface element;

disabling access to a functional user interface element; and

changing an appearance of a functional user interface element; and

wherein the user interface configuration module selects at least one of the func-
tions to configure the user interface of the software application and the
user interface of the operating system.

64. (Currently amended) The system of claim 61, wherein the user interface configuration module ~~performs at least one selected from the group consisting of~~ comprises
program code for performing the functions of:

enabling access to a command;

disabling access to a command;

changing an appearance of a command;

enabling access to a menu;
disabling access to a menu;
changing an appearance of a menu;
enabling access to a button;
disabling access to a button;
changing an appearance of a button;
enabling access to a shortcut; and
disabling access to a shortcut; and

wherein the user interface configuration module selects at least one of the functions to configure the user interface of the software application and the user interface of the operating system.

65. (Cancelled).

66. (Original) The system of claim 61, wherein the user proficiency level detector and the user interface configuration module operate responsive to a trigger event.

67. (Currently amended) The system of claim 61, wherein the user proficiency level ~~detector~~ determiner and the user interface configuration module operate periodically.

68. (Currently amended) The system of claim 61, wherein the user proficiency level ~~detector~~ determiner reads a stored user proficiency level derived from at least one marker.

69. (Currently amended) The system of claim 61, wherein:

the user proficiency level ~~detector detects~~ determiner determines the user proficiency level with respect to a user interface component less than the entire user interface; and

the user interface configuration module automatically configures the at least one functional component of the user interface component without altering the configuration of the remainder of the user interface.

70. (Currently amended) The system of claim 61, wherein:

the user proficiency level ~~detector detects~~ determiner determines the user proficiency level with respect to an application; and

the user interface configuration module automatically configures the at least one functional component of the user interface for the application.

71. (Currently amended) The system of claim 61, further comprising:

a marker storage device, for, responsive to user behavior with respect to the user interface, storing a marker indicating a user proficiency level;

wherein the user proficiency level ~~detector~~ determiner reads the stored marker from the marker storage device.

72. (Currently amended) The system of claim 61, wherein the user proficiency level ~~detector~~ determiner retrieves a plurality of stored markers and aggregates the retrieved markers to derive a proficiency level.

73. (Original) The system of claim 61, further comprising:

a marker storage device, for, responsive to user behavior with respect to the user interface, storing a plurality of markers;
wherein the user proficiency level detector retrieves at least a subset of the stored markers and aggregates the retrieved markers to derive a proficiency level.

74. (Currently amended) The system of claim 61, wherein:

the user proficiency level ~~detector detects~~ determiner determines a user proficiency level with respect to a user interface of a web-resident application being run from a client machine; and
the user interface configuration module automatically configures at least one functional user interface element for the web-resident application.